On-chip Monitoring Infrastructures and Strategies for Many-core Systems

Russell Tessier, Jia Zhao, Justin Lu, Sailaja Madduri, and Wayne Burleson

Research supported by the Semiconductor Research Corporation
Outline

• Motivation
• Contributions
• On-chip monitoring infrastructures
• Extensions to 3D architectures
• Monitoring for voltage droop prevention
• Conclusion and future work
On-chip Sensors and Monitoring

- Thermal sensors, processor activity monitors, delay monitors, reliability monitors, et al.

- Challenges involved
  1. Sensor design (VLSI design research)
  2. Sensor data collection
  3. Sensor data processing, e.g. identify imprecise sensor measurements
  4. Remediation strategies
Multi-core and Many-core Systems

- From single core systems to multi-cores and many-cores
- Need to monitor system temperature, supply voltage fluctuation, reliability, among others
- Remediation strategies include voltage change, frequency change, error protection, et al.

AMD FX-8150 8-Core Bulldozer Processor

Image courtesy: silentpcreview.com
System-Level Requirements

- Monitor data collected in a scalable, coherent fashion
  - **Interconnect flexibility**: Many different monitor interfaces, bandwidth requirements
  - **Low overhead**: Interconnect should provide low overhead interfaces (buses, direct connects) while integrating NoC
  - **Low latency**: Priority data needs immediate attention (thermal, errors)
- Collate data with centralized processing
  - Focus on collaborative use of monitor data
- Validate monitoring system with applications
  - Layout and system-level simulation
MNoC Overview

- A dedicated infrastructure for on-chip monitoring – Monitor Network-on-Chip (MNoC)
- An on-chip network to connect all sensors to a monitor executive processor (MEP) through MNoC routers
- Low latency for sensor data transmission and low cost
Priority Based Interfacing

- Multiple transfer channels available for critical data
- Interface synchronized with router via buffer
- Time stamps used to prioritize/coordinate responses
MNoC Router and Monitor Interface

- Bus and multiplexer are both supported
- Data type:
  - Periodically sampled
  - Occasionally reported
- Critical monitor data is transferred in the priority channel

- A thermal monitor [1] interface example:
  - FSM controlled
  - Time stamp attached to identify out-of-data data

MNoC area with differing router parameters

- Desirable to minimize interconnect data width to just meet latency requirements
- Most of router area consumed by data path
  - Each delay monitor generates 12 bit data + 6 bit header
• Regular channel latency (e.g. 100 cycles) tolerable for low priority data
• Priority channel provides fast path for critical data (20 cycles)
NoC and MNoC Architectures

- A shared memory multicore based on Tile64 [1] and TRIPS OCN [2]
  - 4×4 mesh as in Tile64
  - 256 bit data width
  - 2 flits buffer size
- Monitors in the multicore system
  - Thermal monitor (1/800 injection rate)
  - Delay monitor (around 1/200 injection rate)
- MNoC configuration from the suggested design flow
  - 4×4 MNoC
  - 24-bit flit width
  - 2 virtual channels
  - 2 flits buffer size

MNoC vs. Regular Network-on-chip

- Network-on-chip (NoC) is used in multi-cores
- Why bother to use MNoC for monitor data?
- Three cases
  - NoC with MNoC – monitor data is transferred via MNoC. Inter-processor traffic (“application data”) in the multi-core NoC.
  - MT-NoC (mixed traffic NoC) – All monitor and application traffic in four virtual channel (VC) NoC.
  - Iso-NoC (isolated channel NoC) – monitor data in one VC. Application traffic in the remaining three VCs.
- Virtual channel: another lane on a road. Unfortunately only one lane in the intersections
Three Cases for Comparison

1. **NoC with MNoC** – monitor data is transferred via MNoC. Inter-processor traffic (“application data”) in the multicore NoC.

2. **MT-NoC (mixed traffic NoC)** – All monitor and application traffic in four virtual channel (VC) NoC.

3. **Iso-NoC (isolated channel NoC)** – monitor data in one VC. Application traffic in the remaining three VCs.

- Infrastructure for MT-NoC and Iso-NoC shown
1. NoC with MNoC: lowest latency
2. Iso-NoC: highest latency
3. MT-NoC: lower latency than case 3

- A standalone MNoC ensures low latency for application data
Monitor Data Latency

- Iso-NoC achieves low monitor data latency but has high application data latency
- A standalone MNoC ensures low latency for monitor data
- Modified Popnet network simulator
New On-chip Monitoring Requirement in Many-cores

- Many-core systems demand higher sensor data collecting and processing capability
- New remediation strategies in both local and global scales
  - Signature sharing for voltage droop compensation in the global scale
  - Distributed and centralized dynamic thermal management (DTM)
- Three-dimensional (3D) systems add more complexity
  - Stacking memory layers on top of a core layer

- No previous on-chip monitoring infrastructures can address all these requirements
  - Simple infrastructures based on buses are not suitable
  - The MNoC infrastructure has no support for communications between MEPs
  - Scaled to many-core systems with hundreds to a thousand cores?
  - Support for 3D systems?
3D Many-core Systems

Fig. 1: A three layer 3D system example. Thermal TSVs are for heat dissipation

- 3D technology stacks dies on top of each other
- Through silicon via (TSV) used for vertical communications or heat dissipation
- High bandwidth and high performance

A Hierarchical Sensor Data Interconnect Infrastructure

- An example for a 36 core system
- One sensor NoC router per core
- Sensors are connected to sensor routers (similar to MNoC)
- Sensor routers send data to sensor data processors (SDPs)
  - Through the SDP routers
  - One SDP per 9 cores in this example, may not be the optimal configuration
A Hierarchical Sensor Data Interconnect Infrastructure (cont)

- SDP routers are connected by another layer of network, SDP NoC
- More traffic pattern are supported in the SDP NoC
- Both the sensor NoC and the SDP NoC have low cost
  - Small data width (e.g. 24 bits)
  - Shallow buffers (4-8 flits)
Hierarchical Sensor Data Processing Infrastructure for a 256-core 3D System

- Thermal sensors in the memory layer are connected to sensor routers using through silicon via (TSVs)
- One SDP per 64 cores
SDP Router Design

- SDP routers received packets from sensor routers
- SDP routers also support broadcast
  - Send broadcast packets to SDP
  - Generate new broadcast packets when necessary
- Two virtual channels supported
Packet Transmission in the SDP NoC

- Traffic in the sensor NoC is similar to MNoC
- SDP NoC supports more complicated traffic patterns
  - Hotspot, a global scale DTM [4]
  - Broadcast, a voltage droop signature sharing method [5]
- Hotspot traffic is supported by most routing algorithms
- A SDP router design that supports a simple broadcast strategy with a broadcast controller
  - Send packet vertically first
  - Then send horizontally

Experimental Approach

• Our infrastructure is simulated using a heavily modified Popnet simulator
• Simulated for 256, 512 and 1024 core systems
  – Packet transmission delay
• Synthesized using 45 nm technology
  – Hardware cost
• On-chip sensors
  – Thermal sensors
  – Performance counters
  – Voltage droop sensors
  – Signature-based voltage droop predictors
• A system level experiment is performed using the modified Graphite many-core simulator
  – Run-time temperature is modeled
Modified Graphite Simulator

- Simulation speed is the key due to the large number of cores
- Graphite from the Carbon Research Group at MIT
- Graphite maps each thread in the application to a tile of the target architecture
- These threads are distributed among multiple host processes which are running on multiple host machines
- Mean slowdown vs. native execution around 1000x
- SPLASH2 and PARSEC benchmark supported
- Power model under development

Graphite simulator overview

Image courtesy: J. Miller, etc., “Graphite: A Distributed Parallel Simulator for Multicores”, in Proc. of IEEE International Symposium on High-Performance Computer Architecture (HPCA), Jan 2010
Graphite Initial Experiments

- Graphite compiled and runs on a Core2 Quad Core and 4GB memory machine
- SPLASH2 benchmarks tested with up to 128 cores (1024 next)
- Modification for thermal testing and power evaluation underway
- Integration with modified sensor data network simulator
Experimentations using Graphite

- **3D architecture simulation**
  - Memory blocks stacked up
  - 3D memory latency numbers
  - Adopted in Graphite

- **Integrated with the on-chip sensor interconnect simulator**
  - Thermal, activity, etc. information extraction or estimation using Graphite
  - Data collection and processing using a modified Popnet
  - Remediation simulation

- **Dynamic frequency and voltage scaling experimentations in Graphite**
  - Change simulation parameter at run-time
Comparison against a Flat Sensor NoC Infrastructure

<table>
<thead>
<tr>
<th>Core and SDP num.</th>
<th>Latency type</th>
<th>Flat sensor NoC (cycles)</th>
<th>Our method (cycles)</th>
<th>Latency reduction w.r.t. flat sensor NoC (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 core (4 SDP)</td>
<td>Inter-SDP</td>
<td>45.43</td>
<td>7.85</td>
<td>82.72</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>62.82</td>
<td>25.24</td>
<td>59.82</td>
</tr>
<tr>
<td>512 core (8 SDP)</td>
<td>Inter-SDP</td>
<td>67.57</td>
<td>11.37</td>
<td>83.17</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>84.96</td>
<td>28.76</td>
<td>66.15</td>
</tr>
<tr>
<td>1024 core (16 SDP)</td>
<td>Inter-SDP</td>
<td>90.36</td>
<td>14.38</td>
<td>84.08</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>107.75</td>
<td>31.77</td>
<td>70.52</td>
</tr>
</tbody>
</table>

- Simulated using a modified Popnet, one SDP per 64 core is chosen, sensor data from the memory layer included
- Compare the latency of our infrastructure versus a flat sensor NoC infrastructure
  - Only sensor routers, no SDP NoC
  - Packets between SDPs (inter-SDP) are transmitted using sensor routers
- Our infrastructure scientifically reduce the inter-SDP (>82%) and total latency (>59%)
- Hardware cost increase with respect to the flat sensor NoC less than 6%
- Our infrastructure provides higher throughput versus the flat sensor NoC
Core to SDP Ratio Experiment

<table>
<thead>
<tr>
<th>Core num.</th>
<th>Core/SDP ratio</th>
<th>SDP num.</th>
<th>Sensor NoC latency</th>
<th>SDP latency</th>
<th>Total latency</th>
<th>SDP NoC to sensor NoC HW cost ratio (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>32</td>
<td>8</td>
<td>12.25</td>
<td>11.01</td>
<td>23.26</td>
<td>7.94</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>4</td>
<td>17.31</td>
<td>7.72</td>
<td>25.03</td>
<td>4.19</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>2</td>
<td>24.14</td>
<td>5.25</td>
<td>29.39</td>
<td>1.72</td>
</tr>
<tr>
<td>512</td>
<td>32</td>
<td>16</td>
<td>12.25</td>
<td>13.37</td>
<td>25.62</td>
<td>8.78</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>8</td>
<td>17.31</td>
<td>11.42</td>
<td>28.73</td>
<td>4.99</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>4</td>
<td>24.14</td>
<td>7.88</td>
<td>32.02</td>
<td>2.65</td>
</tr>
<tr>
<td>1024</td>
<td>32</td>
<td>32</td>
<td>12.25</td>
<td>20.87</td>
<td>33.12</td>
<td>9.20</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>16</td>
<td>17.31</td>
<td>15.48</td>
<td>32.79</td>
<td>5.46</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>8</td>
<td>24.14</td>
<td>11.84</td>
<td>35.98</td>
<td>3.15</td>
</tr>
</tbody>
</table>

- One SDP per 64 cores is chosen
  - Low latency
  - Moderate hardware cost, less than 6% versus only the sensor NoC
Throughput Comparison

- Compare the throughput of the inter-SDP packet transmission
  - Same throughput for packet transmission in the sensor NoC
- Our infrastructure provides higher throughput versus the flat sensor NoC
Signature based Voltage Droop Compensation

- Event history table content is compared with signatures at run-time to predict emergencies
  - Large table -> more accurate prediction
- Signature table
  - Larger table -> higher performance
  - Larger table -> higher cost
- Extensively studied in Reddi, et al [1]

A Thermal-aware Voltage Droop Compensation Method with Signature Sharing

- High voltage droops cause voltage emergencies
- Voltage droop prediction based on signatures
  - Signatures: footsteps of a serial of instructions
  - Prediction of incoming high voltage droops
- Signature based method in single core systems
- Initial signature detection involves performance penalty

**Idea 1:** Signature sharing across cores
- Fewer penalties for initial signature detection
Voltage Droop Signature Sharing in Multicore Systems

- 8 and 16 core systems simulated by SESC, 8 core results shown here
- Comparison between the signature sharing method and the local signature method
- Four benchmarks show significant reduction in signature number
- Performance benefit mainly come from lesser rollback penalty

### Sign. Num. reduction (%)

<table>
<thead>
<tr>
<th>Test bench</th>
<th>Case</th>
<th>Sign. Num.</th>
<th>Sign. Num. reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water-spatial</td>
<td>Local only</td>
<td>18,474</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>5,167</td>
<td></td>
</tr>
<tr>
<td>Fmm</td>
<td>Local only</td>
<td>832</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>202</td>
<td></td>
</tr>
<tr>
<td>LU</td>
<td>Local only</td>
<td>40,838</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>15,655</td>
<td></td>
</tr>
<tr>
<td>Ocean</td>
<td>Local only</td>
<td>271,179</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>224,151</td>
<td></td>
</tr>
</tbody>
</table>

### Exec. Time reduction (%)

<table>
<thead>
<tr>
<th>Test bench</th>
<th>Case</th>
<th>Exec. Time (ms)</th>
<th>Exec. time reduction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water-spatial</td>
<td>Local only</td>
<td>14.59</td>
<td>2.44</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>14.23</td>
<td></td>
</tr>
<tr>
<td>Fmm</td>
<td>Local only</td>
<td>10.13</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>10.13</td>
<td></td>
</tr>
<tr>
<td>LU</td>
<td>Local only</td>
<td>17.20</td>
<td>4.48</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>16.43</td>
<td></td>
</tr>
<tr>
<td>Ocean</td>
<td>Local only</td>
<td>25.90</td>
<td>5.57</td>
</tr>
<tr>
<td></td>
<td>Global sharing</td>
<td>24.46</td>
<td></td>
</tr>
</tbody>
</table>
A Thermal-aware Voltage Droop Compensation Method with Signature Sharing (cont)

- Reduce system frequencies to combat high voltage droops
  - Previous research considers only one reduced frequency
- Our experiment show that voltage droop decreases as temperature increases with the same processor activity
- **Idea 2**: Choose different reduced frequencies according to temperature

![Graph showing the decrease in voltage droop with increasing temperature.](image)

- Approximately 0.22% decrease in peak-droop per 10°C rise in temperature.
Thermal-aware Voltage Compensation Method for Multi-core Systems

- 5 frequency cases, normal frequency is 2GHz
- Case 1 uses one reduced frequency
- Cases 2, 3 and 4 best show the performance benefits of the proposed method
- Performance benefits 5% on average (8 core and 16 core systems) [9]

<table>
<thead>
<tr>
<th>Temp. range</th>
<th>Frequency table case</th>
</tr>
</thead>
<tbody>
<tr>
<td>20-40°C</td>
<td>1GHz 1GHz 1GHz 1GHz 1GHz</td>
</tr>
<tr>
<td>40-60°C</td>
<td>1GHz 1GHz 1GHz 1.3GHz 1.3GHz</td>
</tr>
<tr>
<td>60-80°C</td>
<td>1GHz 1.3GHz 1.3GHz 1.3GHz 1.3GHz</td>
</tr>
<tr>
<td>80-100°C</td>
<td>1GHz 1.3GHz 1.3GHz 1.3GHz 1.3GHz</td>
</tr>
</tbody>
</table>

A System Level Experiment Result

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Performance (billion cycles)</th>
<th>Benefit (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Case 1</td>
<td>Case 2</td>
</tr>
<tr>
<td>LU (contig)</td>
<td>24.23</td>
<td>23.70</td>
</tr>
<tr>
<td>Ocean (contig)</td>
<td>2.76</td>
<td>2.54</td>
</tr>
<tr>
<td>Radix</td>
<td>9.78</td>
<td>9.29</td>
</tr>
<tr>
<td>FFT</td>
<td>115.14</td>
<td>115.06</td>
</tr>
<tr>
<td>Cholesky</td>
<td>189.66</td>
<td>185.05</td>
</tr>
<tr>
<td>Radiosity</td>
<td>121.42</td>
<td>114.71</td>
</tr>
</tbody>
</table>

- A 128-core 3D system with 2 layers simulated using a modified Graphite
- Use dynamic frequency scaling (DFS) for thermal management and voltage droop compensation
  - Case 1: DFS for thermal management only
  - Case 2: DFS for voltage droop, using the flat sensor NoC
  - Case 3: DFS for voltage droop, using our hierarchical infrastructure
Conclusion

- On-chip monitoring of temperature, performance, supply voltage and other environmental conditions
- New infrastructures for on-chip sensor data processing for multi-core and many-core systems
  - The MNoC infrastructure
  - A hierarchical infrastructure for many-core systems. Significant latency reduction (>50%) versus a flat sensor NoC
- New remediation strategies using dedicated on-chip monitoring infrastructures
  - A thermal-aware voltage droop compensation method with signature sharing. Performance benefit 5% on average
- Other monitoring efforts
  - Sensor calibration
Publications


